TechnoBuzz
@CSE

Editorial Team

FACULTY

MR. VIJAY C.P
Assistant Professor
Dept. of CSE

MR. HARISHA
Assistant Professor
Dept. of CSE
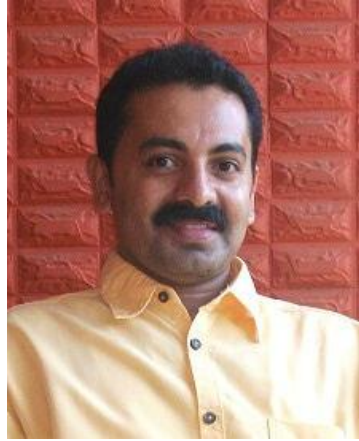
STUDENTS

RUMAAN KHALANDER
Third Year
Dept. of CSE

MUSTHAQ AHMED
Third Year
Dept. of CSE

JEHAD MOHAMED
Second Year
Dept. of CSE

A word from the HOD's desk

Dear Reader,

Greetings.

I feel privileged in presenting our department tech magazine "TECHNOBUZZ@CSE", 2017 Edition , it is designed to showcase trending technologies and events and the tech talents of our students and faculty members.

With a sense of pride and satisfaction, I would like to say with the active support of management, Director, Principal faculty members and students, the department tech magazine has come alive .

I extend my thanks to my students and colleagues for being a part of the editorial board. With all the efforts and contributions put in by the faculty members and students; This would definitely create an impact in the minds of readers, by way of providing larger visibility and dimension to department.

PROF. SUDHEER SHETTY

HOD, CSE DEPT.

Vision

To be a globally recognized center for imparting quality technical education through innovative research and incubation with moral values in the field of computer science and engineering.

Mission

M1: Inculcating quality teaching learning process by creating in-house industry-institute interactions to meet the societal needs.

M2: Developing research and entrepreneurship culture in students and faculty to cultivate lifelong learning.

M3: Empowering professionals with strong ethical values and leadership abilities.

Programme Educational Objectives (PEO's)

- Engage in solving real life problems by applying the gained technical knowledge and lifelong learning.
- To Encourage students for higher studies and indulge in research activities.
- To Excel as a team leader, entrepreneur and a member in multidisciplinary environment.
- Ability to apply modern tools, skills and techniques to develop software and hardware systems to meet the need of the modern society.
- Practice moral and ethical values to solve environmental and societal issues in the field of computer science and engineering.

06

7 Best Programming Languages to Learn in 2017
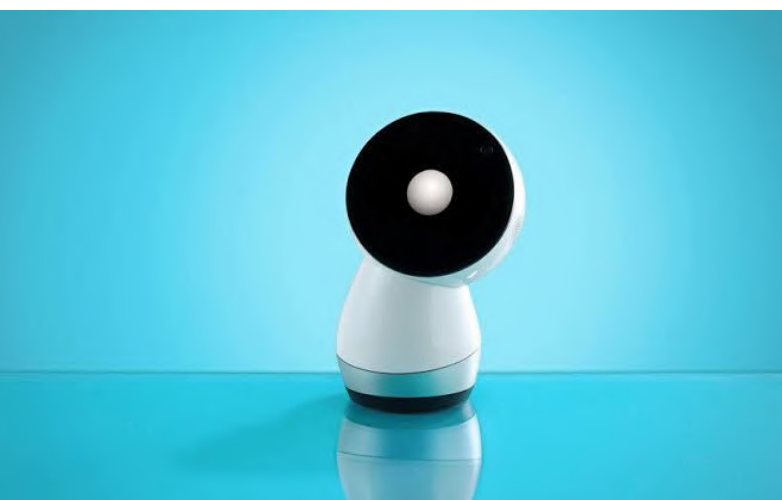
Brace yourselves. JS is coming.

08

Ransomware WannaCry : All you need to know

What is a ransomware? What is WannaCry?
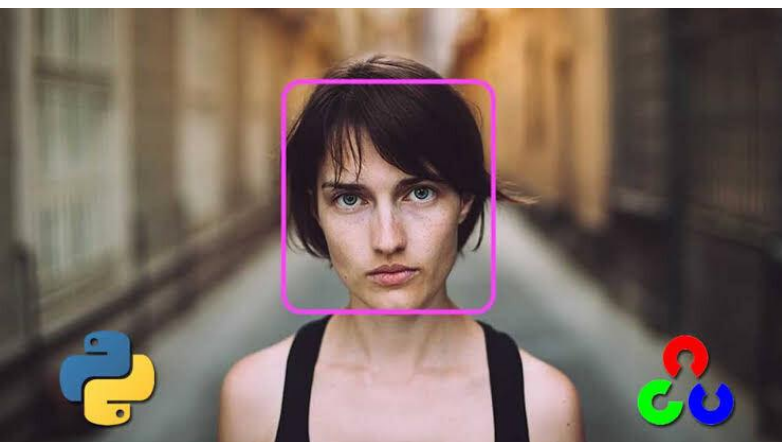
09

A Robot You Can Relate To - Jibo

Jibo seems downright human in a way that his predecessors do not.

10

Google X and the Science of Radical Creativity

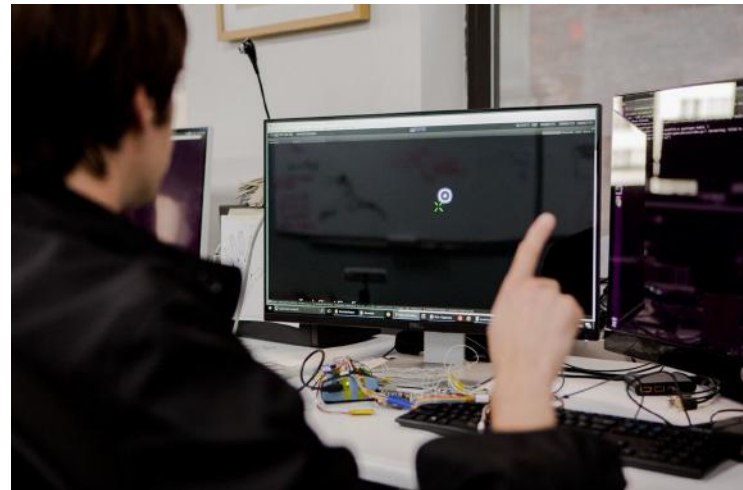How the secretive Silicon Valley lab is trying to resurrect the lost art of invention

12

Brain-Machine Interface Isn't Sci-Fi Anymore

This startup lets you control machines with your mind—no implants required.

13

Face Detection in 5 Minutes using OpenCV & Python

Yes. 5 minutes.

7 Best Programming Languages to Learn in 2017

BY RUMAAN KHALANDER

We conducted our own survey & researched the most popular programming languages. We conclude with the following programming languages for 2017:

JavaScript

JavaScript – The most popular one. Usersnap stated in the article on the best web development trends for 2017 that the most commonly used programming language is JavaScript. The latest version of JavaScript (ES2017) is launching in late 2017 and developers are already emotional about it.

If you want to take a deep dive, Douglas Crockford's "JavaScript – The Good Parts" will walk you through all the peculiarities of JavaScript.

Java

Java – The oldie, but goldie. No list is complete without Java. In the long run, it's always a great choice & the stats suggest it's not going away anytime soon.Java is used on 15 billion (that's not a typo) devices and over 10 million developers use Java worldwide!Learn Java if you create Android apps, games, software and website content. Example sites that use Java are Amazon, LinkedIn, plus eBay. Java 9 is launching in 2017. So definitely check it out when it's ready.

Python

Python – The standard one. Python is an object-oriented language that closely resembles the English language which makes it a great language to learn for beginners as well as seasoned professionals. Examples sites that use Python are Instagram, YouTube, Reddit, NASA, and Usersnap (who wrote about their Python experience here) Python 3.6 got released in December 2016 with some awesome features.

Elixir

Elixir – The unknown one. Elixir is a functional, dynamic language created for building scalable, maintainable apps. Concurrency is one of its main benefits. It's great for large apps that handle a lot of tasks simultaneously.Example sites that use Elixir are Pinterest, Moz etc.

Rust

Rust – The beloved one. Rust is the most loved programming language on StackOverflow for 2016 which says a lot.It's a general-purpose language to create fast, secure applications which take advantage of the powerful features of modern multi-core processors.Example sites that use Rust are Dropbox or Coursera.

Go

Go – The googly one. Go (or GOLANG) – by Google – is going to grow in popularity in 2017.It has an excellent standard library and it compiles fast. It's great with concurrent tasks and programs as well.Example sites that use Go are Netflix, YouTube, and Adobe.

TypeScript

TypeScript – Microsoft's answer to JavaScript. TypeScript is a statically typed language that compiles to JavaScript and it's growing fast (see Google Trends below)! The new version 2.1 includes all the new features of JavaScript with optional static types.

Ransomware WannaCry : All you need to know

BY MUSTHAQ AHMED

WannaCry is an example of crypto ransomware, a type of malicious software (malware) used by cybercriminals to extort money.Ransomware does this by either encrypting valuable files, so you are unable to read them, or by locking you out of your computer, so you are not able to use it.Ransomware that uses encryption is called crypto ransomware. The type that locks you out of your computer is called locker ransomware.Like other types of crypto-ransomware, WannaCry takes your data hostage, promising to return it if you pay a ransom.WannaCry targets computers using Microsoft Windows as an operating system. It encrypts data and demands payment of a ransom in the cryptocurrency Bitcoin for its return.
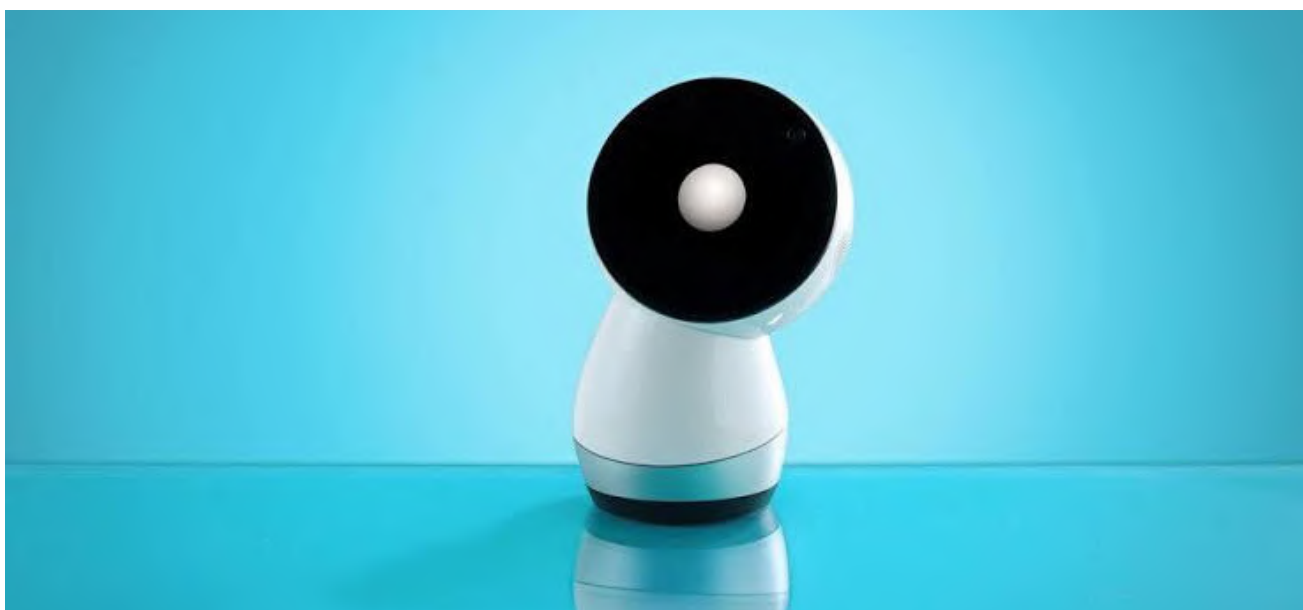
What was the WannaCry ransomware attack? The WannaCry ransomware attack was a global epidemic that took place in May 2017.This ransomware attack spread through computers operating Microsoft Windows. User's files were held hostage, and a Bitcoin ransom was demanded for their return.Were it not for the continued use of outdated computer systems and poor education around the need to update software, the damage caused by this attack could have been avoided.The WannaCry ransomware attack hit around 230,000 computers globally.One of the first companies affected was the Spanish mobile company, Telefónica. By May 12th, thousands of NHS hospitals and surgeries across the UK were affected.A third of NHS hospital trusts were affected by the attack. Terrifyingly ambulances were reportedly rerouted, leaving people in need of urgent care in need. It was estimated to cost the NHS a whopping £92 million after 19,000 appointments were canceled as a result of the attack.As the ransomware spread beyond Europe, computer systems in 150 countries were crippled. The WannaCry ransomware attack had a substantial financial impact worldwide. It is estimated this cybercrime caused $4 billion in losses across the globe.

A Robot You Can Relate To - Jibo

BY SHREYAS BALIGA

Personal robots, such as Amazon Echo and Google Home, have come a long way in recent years. But fundamentally, they're still stationary speakers whose defining expression is a light that turns on when you speak.Jibo is different. It's not just that he—and I use the term he here, because that's how Jibo refers to himself—looks like something straight out of a Pixar movie, with a big, round head and a face that uses animated icons to convey emotion. It's not just that his body swivels and swerves while he speaks, as if he's talking with his nonexistent hands. It's not just that he can giggle and dance and turn to face you, wherever you are, as soon as you say, "Hey, Jibo."

It's that, because of all this, Jibo seems downright human in a way that his predecessors do not.And while that technology may seem merely amusing—or creepy, depending on your point of view—it could fundamentally reshape how we interact with machines.Jibo still has a lot to learn. Although he can help users in basic ways, like by summarizing news stories and taking photos, he can't yet play music requests or work with third-party apps like Domino's and Uber, which, at $899, could make him a tough sell. But Matt Revis, the company's vice president of product management, is confident Jibo will evolve. "There was a threshold we had to reach in order to launch," says Revis. "Now it's part of the journey."

Google X and the Science of Radical Creativity

BY CHAMAN K

The setting is X, the so-called moonshot factory at Alphabet, the parent company of Google. And the scene is not the beginning of some elaborate joke. The people in this room have a particular talent: They dream up far-out answers to crucial problems. The dearth of housing in crowded and productive coastal cities is a crucial problem.

Oceanic residences are, well, far-out. At the group's invitation, I was proposing my own moonshot idea, despite deep fear that the group would mock it.Like a think-tank panel with the instincts of an improv troupe, the group sprang into an interrogative frenzy. "What are the specific economic benefits of increasing housing supply?" the liquid-crystals guy asked. "Isn't the real problem that transportation infrastructure is so expensive?" the balloon scientist said. "How sure are we that living in densely built cities makes us happier?" the extradimensional physicist wondered. Over the course of an hour, the conversation turned to the ergonomics of Tokyo's high-speed trains and then to Americans' cultural preference for suburbs.

Members of the team discussed commonsense solutions to urban density, such as more money for transit, and eccentric ideas, such as acoustic technology to make apartments soundproof and self-driving housing units that could park on top of one another in a city center. At one point, teleportation enjoyed a brief hearing.

X is perhaps the only enterprise on the planet where regular investigation into the absurd is not just permitted but encouraged, and even required. X has quietly looked into space elevators and cold fusion.

It has tried, and abandoned, projects to design hoverboards with magnetic levitation and to make affordable fuel from seawater. It has tried—and succeeded, in varying measures—to build self-driving cars, make drones that deliver aerodynamic packages, and design contact lenses that measure glucose levels in a diabetic person's tears.These ideas might sound too random to contain a unifying principle. But they do. Each X idea adheres to a simple three-part formula. First, it must address a huge problem; second, it must propose a radical solution; third, it must employ a relatively feasible technology. In other words, any idea can be a moonshot—unless it's frivolous, small-bore, or impossible.
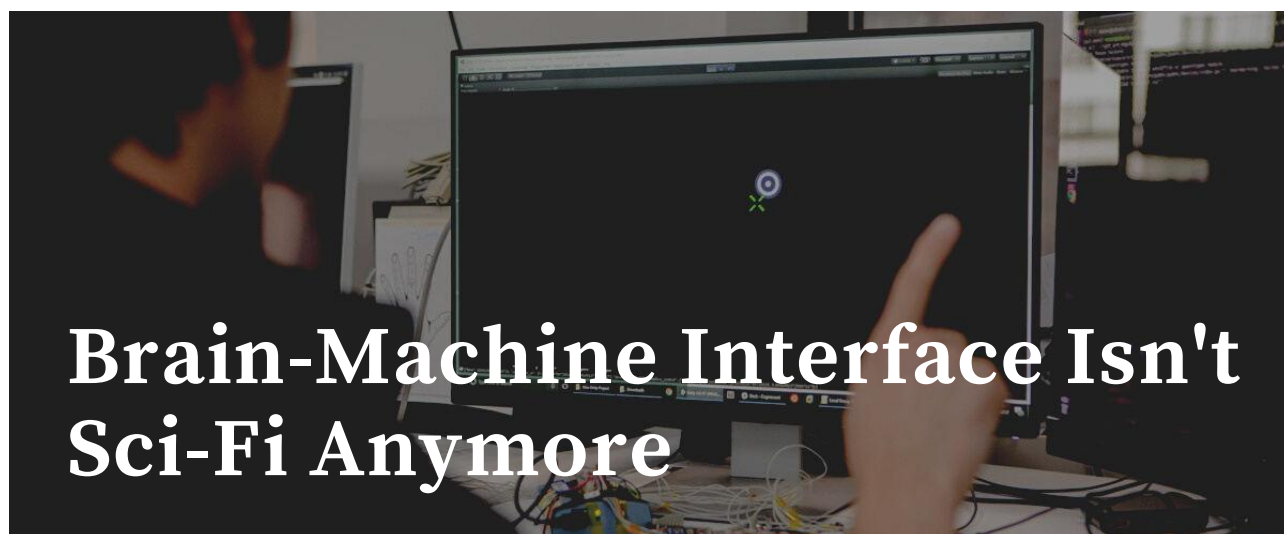
Brain-Machine Interface Isn't Sci-Fi Anymore

Thomas Reardon puts a terrycloth stretch band with microchips and electrodes woven into the fabric—a steampunk version of jewelry—on each of his forearms. "This demo is a mind fuck," says Reardon, who prefers to be called by his surname only. He sits down at a computer keyboard, fires up his monitor, and begins typing. After a few lines of text, he pushes the keyboard away, exposing the white surface of a conference table in the midtown Manhattan headquarters of his startup. He resumes typing. Only this time he is typing on…nothing. Just the flat tabletop. Yet the result is the same: The words he taps out appear on the monitor.

That's cool, but what makes it more than a magic trick is how it's happening. The text on the screen is being generated not by his fingertips, but rather by the signals his brain is sending to his fingers. The armband is intercepting those signals, interpreting them correctly, and relaying the output to the computer, just as a keyboard would have.

Whether or not Reardon's digits actually drum the table is irrelevant—whether he has a hand is irrelevant—it's a loop of his brain to machine. What's more, Reardon and his colleagues have found that the machine can pick up more subtle signals—like the twitches of a finger—rather than mimicking actual typing.

You could be blasting a hundred words a minute on your smart phone with your hands in your pockets. In fact, just before Reardon did his mind-fuck demo, I watched his cofounder, Patrick Kaifosh, play a game of Asteroids on his iPhone. He had one of those weird armbands sitting between his wrist and his elbows. On the screen you could see Asteroids as played by a decent gamer, with the tiny spaceship deftly avoiding big rocks and spinning around to blast them into little pixels. But the motions Kaifosh was making to control the game were barely perceptible: little palpitations of his fingers as his palm lay flat against the tabletop. It seemed like he was playing the game only with mind control. And he kind of was.

Face Detection in 5 Minutes using OpenCV & Python

First of all make sure you have OpenCV installed. You can install it using pip.

```
pip install opencv-python
```

Face detection using Haar cascades is a machine learning based approach where a cascade function is trained with a set of input data. OpenCV already contains many pre-trained classifiers for face, eyes, smiles, etc.. Today we will be using the face classifier. You can experiment with other classifiers as well.You need to download the trained classifier XML file (haarcascade_frontalface_default.xml), which is available in OpenCv's GitHub repository. Save it to your working location.To detect faces in images:

```python
import cv2

# Load the cascade
face_cascade = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')
# Read the input image
img = cv2.imread('test.jpg')
# Convert into grayscale
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
# Detect faces
faces = face_cascade.detectMultiScale(gray, 1.1, 4)
# Draw rectangle around the faces
for (x, y, w, h) in faces:
    cv2.rectangle(img, (x, y), (x+w, y+h), (255, 0, 0), 2)
# Display the output
cv2.imshow('img', img)
cv2.waitKey()
```

And thats it. Make sure you specify the proper input image as png or jpg .